## REMARKS

Claims 1 to 17 are currently pending in the present application.

It is respectfully submitted that all of the presently pending claims are allowable, and reconsideration of the present application is respectfully requested.

Claims 1 to 8, 10, 11, 13, and 14 were rejected under 35 U.S.C. § 102(b) as anticipated by U.S. Patent No. 5,680,620 ("Ross").

As regards the anticipation rejections of the claims, to reject a claim under 35 U.S.C. § 102, the Office must demonstrate that each and every claim feature is identically described or contained in a single prior art reference. (See *Scripps Clinic & Research Foundation v. Genentech, Inc.*, 18 U.S.P.Q.2d 1001, 1010 (Fed. Cir. 1991)). As explained herein, it is respectfully submitted that the prior Office Action does not meet this standard, for example, as to all of the features of the claims. Still further, not only must each of the claim features be identically described, an anticipatory reference must also enable a person having ordinary skill in the art to practice the claimed subject matter. (See *Akzo, N.V. v. U.S.I.T.C.*, 1 U.S.P.Q.2d 1241, 1245 (Fed. Cir. 1986)).

As further regards the anticipation rejections, to the extent that the Office Action may be relying on the inherency doctrine, it is respectfully submitted that to rely on inherency, the Office must provide a "basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic *necessarily* flows from the teachings of the applied art." (See M.P.E.P. § 2112; emphasis in original; and see *Ex parte Levy*, 17 U.S.P.Q.2d 1461, 1464 (Bd. Pat. App. & Int'f. 1990)). Thus, the M.P.E.P. and the case law make clear that simply because a certain result or characteristic may occur in the prior art does not establish the inherency of that result or characteristic.

Claim 1 relates to a computer readable medium having a program, the program performing a method for monitoring an execution of another program that is executable on at least one microprocessor of a micro controller using a debug logic of the micro controller, the method including the features of *causing the debug logic to trigger an exception* upon access to a specific address range during a program execution time, and *causing the debug logic to execute an exception routine* after the exception is triggered during the program execution time, in which *the access to the specific address range includes access to an illegal storage area*, in which *the debug logic and its registers are operated in parallel to the program*

*execution time . . . , so as to provide a secure stack check without using the program execution time of the microprocessor*, and in which *the debug logic monitors a program run.*

It is respectfully submitted that the Ross reference does not identically disclose (or even suggest) all of the features of claim 1. The Ross reference merely refers to a processor having a debug register, in which the processor notifies one program of another program's access to a shared resource. (Ross, col. 2, lines 14 to 24). In this regard, the debug register of Ross is merely a *storage block*, which is indicated as an integral component of the processor. (Ross, col. 2, line 67, to col. 3, line 2; and col. 4, lines 3 to 36). In short, the debug register of Ross is merely a storage block.

In contrast, claim 1 of the present application includes the feature of *causing the debug logic to trigger an exception. Claim 1 clearly provides that the debug logic -- and not the processor -- triggers an exception.* Further, the Specification describes the triggering of an exception as "particularly an interrupt of the program execution." (Specification, p. 8, line 28). The Ross reference, however, explicitly states that "[w]hen processor 12 detects [the breakpoint] register being set, then ***processor 12 generates*** *a device access interrupt.*" (Ross, col. 5, lines 1 to 3 (emphasis added)). As to the text at column 4, lines 37 to 56 of the Ross reference, the cited section does not identically disclose (or even suggest) that the debug register circuit of Ross triggers an exception. Thus, Ross makes plain that its processor -- and not its debug register -- triggers an exception. Therefore, Ross does not identically disclose (or even suggest) the feature of *causing the debug logic to trigger an exception*, as provided for in the context of claim 1.

Claim 1 of the present application also includes the feature of *causing the debug logic to execute an exception routine.* Claim 1 clearly provides that the debug logic -- and not the processor -- executes an exception routine. The Ross reference, however, explicitly states that "***processor 12 executes*** *a debug interrupt service routine.*" (Ross, col. 5, lines 64 to 65 (emphasis added)). As to the text at column 4, lines 37 to 56 of Ross, it only refers to "executing an exceptional routine after the exception is triggered", so that it does not disclose that the debug register circuit of Ross executes an exception routine. That is, Ross makes plain that its processor -- and not its debug register -- executes an exception routine. Therefore, Ross does not identically disclose (or even suggest) the feature of *causing the debug logic to execute an exception routine*, as provided for in the context of claim 1.

In addition, claim 1 of the present application includes the feature in which *the access to the specific address range includes access to an illegal storage area.* The present

Specification specifically defines illegal storage areas as "storage areas which are physically not present or which lie outside a storage area provided," or which are "beyond a preselectable maximum stack size." (Specification, p. 5, lines 15 to 16; and p. 11, lines 1 to 19).

In stark contrast, Ross does not disclose any such illegal storage area, since it only states that "[t]he address which is set as a ***breakpoint corresponds to the address*** *which is called when access to the device is desired.*" (Ross, col. 4, lines 42 to 44 (emphasis added)). As to the text at column 5, lines 7 to 15 of Ross, it only refers to "accessing to an illegal storage area or protected address", and therefore simply does not identically disclose that the breakpoint of Ross is an illegal storage area. Thus, the breakpoint address of Ross is a valid address of a connected device, and therefore is not an illegal storage area which is either physically not present, lies outside a storage area provided, or is beyond a preselectable maximum stack size, as provided for in the present application and in the context of the claimed subject matter.

Therefore, Ross does not identically disclose (or even suggest) the feature in which *the access to the specific address range includes access to an illegal storage area*, as provided for in the context of claim 1.

Further, claim 1 includes the features in which *the debug logic and its registers are operated in parallel to the program execution time, so as to provide a secure stack check without using the program execution time of the microprocessor.* The present application further discloses that "the debug logic affects neither the computing performance of the microprocessor nor the program memory," and that "reliable functioning of the fault analysis is provided independently of the computing load of the microprocessor." (Specification, p. 5, lines 22 to 23, and lines 27 to 28).

In stark contrast, Ross explicitly states that "the breakpoint register is ***monitored by processor 12***," and "***processor 12 continues to monitor*** the breakpoint register." (Ross, col. 4, lines 57 to 64 (emphases added)). Further, the "***processor 12 generates*** a device access interrupt," and "***processor 12 executes*** a debug interrupt service routine." (Ross, col. 5, lines 2 to 3, and lines 64 to 65 (emphases added)). As to the text at column 4, lines 57 to 66, and column 5, lines 32 to 47, Ross only refers to "monitoring of the program 1 and program 2 are operating in parallel," and the text at column 3, line 65 to column 4, line 2, column 5, lines 6 to 12, and lines 25 to 27 of Ross only states "that the secure stack check is inherently done without using the program execution time of the microprocessor when the debug logic and its

register are operated in parallel to the program execution time." However, nowhere do these cited sections identically disclose (or even suggest) that the debug registers of Ross are operated in parallel to the program execution time of the processor because, as quoted above, the processor of Ross -- and not its debug register -- performs all the functions. Thus, Ross plainly does not identically disclose (or even suggest) a debug logic operated in parallel to the program execution time, without using the program execution time of the microprocessor.

Therefore, Ross does not identically disclose (or even suggest) the features in which *the debug logic and its registers are operated in parallel to the program execution time . . . , so as to provide a secure stack check without using the program execution time of the microprocessor*, as provided for in the context of claim 1.

Moreover, claim 1 of the present application includes the feature in which *the debug logic monitors a program run*. The Specification makes plain that the "debug logic . . . is used for monitoring the program for faults." (Specification, p. 5, lines 10 to 11; and p. 8, lines 24 to 26). In stark contrast, Ross explicitly states that "the breakpoint register is ***monitored by processor 12***," and "***processor 12 continues to monitor*** the breakpoint register." (Ross, col. 4, lines 57 to 64 (emphases added)). Thus, Ross makes plain that its processor -- and not its debug register -- monitors the program for faults. Therefore, Ross does not identically disclose (or even suggest) the feature that *the debug logic monitors a program run*, as provided for in the context of claim 1.

Further, Figures 1 and 2 of the present application plainly show a microprocessor 4 and a separate debug logic 6. In stark contrast, Ross explicitly states that its "notification capability is provided *without the use of extra or dedicated hardware resource, such as a processor interrupt capability*." Accordingly, Ross makes plain that its processor -- and not its debug register, which is merely a storage block -- monitors the program for faults.

Therefore, Ross does not identically disclose (or even suggest) all of the features of claim 1, including the features of *causing the debug logic to trigger an exception* upon access to a specific address range during a program execution time, and *causing the debug logic to execute an exception routine* after the exception is triggered during the program execution time, in which *the access to the specific address range includes access to an illegal storage area*, in which *the debug logic and its registers are operated in parallel to the program execution time . . . , so as to provide a secure stack check without using the program execution time of the microprocessor*, and in which *the debug logic monitors a program run*.

Accordingly, it is respectfully submitted that claim 1 is allowable for at least these reasons. Claims 2 to 8 depend from and claim 1, and are therefore allowable for at least the same reasons as claim 1.

Claim 10 relates to a control element for a micro controller, and includes features like those of claim 1. Accordingly, it is respectfully submitted that claim 10 is allowable for essentially the same reasons as claim 1, as is its dependent claim 11.

Claim 13 relates to a micro controller, including at least one microprocessor and a debug logic, and includes features like those of claim 1. Accordingly, it is respectfully submitted that claim 13 is allowable for essentially the same reasons as claim 1, as is its dependent claim 14.

Withdrawal of these rejections is therefore respectfully requested.

Claims 9, and 15 to 17 were rejected under 35 U.S.C. § 103(a) as unpatentable over Ross in view of U.S. Patent No. 6,535,811 ("Rowland et al.").

To reject a claim under 35 U.S.C. § 103(a), the Office bears the initial burden of presenting a *prima facie* case of obviousness. *In re Rijckaert*, 9 F.3d 1531, 1532, 28 U.S.P.Q.2d 1955, 1956 (Fed. Cir. 1993). To establish *prima facie* obviousness, three criteria must be satisfied. First, there may be some suggestion or motivation to modify or combine reference teachings. *In re Fine*, 837 F.2d 1071, 5 U.S.P.Q.2d 1596 (Fed. Cir. 1988). This teaching or suggestion to make the claimed combination may be found in the prior art and not based on the application disclosure. *In re Vaeck*, 947 F.2d 488, 20 U.S.P.Q.2d 1438 (Fed. Cir. 1991).

As clearly indicated by the Supreme Court, it is "important to identify a reason that would have prompted a person of ordinary skill in the relevant field to combine the [prior art] elements" in the manner claimed. *See KSR Int'l Co. v. Teleflex, Inc.*, 127 S. Ct. 1727 (2007). In this regard, the Supreme Court further noted that "rejections on obviousness cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness." *Id.*, at 1396. Second, there must be a reasonable expectation of success. *In re Merck & Co., Inc.*, 800 F.2d 1091, 231 U.S.P.Q. 375 (Fed. Cir. 1986). Third, the prior art reference(s) must teach or suggest all of the claim features. *In re Royka*, 490 F.2d 981, 180 U.S.P.Q. 580 (C.C.P.A. 1974).

As essentially explained above, Ross does not disclose (or even suggest) all of the features of claims 1, 10, and 13. Further, it is respectfully submitted that even if it were proper to combine Ross and Rowland (which is not conceded), the secondary Rowland reference does not cure -- and is not asserted to cure -- the critical deficiencies of the Ross reference.

Accordingly, dependent claims 9 and 16 are allowable for essentially the same reasons as claim 1, dependent claim 15 is allowable for essentially the same reasons as claim 13, and dependent claim 17 is allowable for essentially the same reasons as claim 10, since the Rowland reference does not cure -- and is not asserted to cure -- the critical deficiencies of the primary Ross reference.

Additionally, each of claims 15, 16 and 17 includes the following further features in which: *a memory address is stored that was accessed before an occurrence of the fault in the fault memory and at least a type of a fault is stored in a fault memory, before the micro controller is reset and started up again and before the program is initialized, during the execution of the exception routine, the micro controller is reset, the micro controller is started up again, and the program is initialized, the exception corresponds to an interrupt of the execution of the program, the debug logic is configured during a startup of the micro controller, the debug logic monitors whether the program one of (i) accesses a preselectable address range of a memory during the program execution time, and (ii) accesses an address range of a stack of the micro controller beyond a preselectable maximum stack size during the program execution time, and the debug logic monitors whether an attempt is made during the program execution time to execute a code sequence of the program in a first type of memory, swapped out from the first type of memory of the micro controller into another type of memory of the microcontroller.* The applied references do not disclose this combination of features, including for the reasons explained as to the respective base claims.

Withdrawal of the obviousness rejections is therefore respectfully requested.

Claim 12 was rejected under 35 U.S.C. § 103(a) as unpatentable over Ross, in view of that which the Office Action characterizes as Admitted Prior Art ("Background Information").

*Applicants first note that the Office Action again relies on the BACKGROUND INFORMATION Section of the present application. In this regard, it is noted that while certain published information may represent prior art (namely, any cited patents that are*

*prior art), the information concerning the "debug logic triggering" may represent internal Bosch information.*

Regardless of the proper characterization of the "Background Information" (which is not conceded to as asserted), as explained above, Ross does not disclose (or even suggest) all of the features of claim 10. Further, the "Background Information" does not cure -- and is not asserted to cure -- the critical deficiencies of the Ross reference.

Accordingly, dependent claim 12 is allowable for essentially the same reasons as claim 10, since the "Background Information" does not cure -- and is not asserted to cure -- the critical deficiencies of the primary Ross reference. Withdrawal of this obviousness rejection is therefore respectfully requested.

*As further regards all of the obviousness rejections, any Official Notice is respectfully traversed to the extent that it is maintained and it is requested that the Examiner provide specific evidence to establish those assertions and/or contentions that may be supported by that Official Notices under 37 C.F.R. § 1.104(d)(2) or otherwise. In particular, it is respectfully requested that the Examiner provide an affidavit and/or that the Examiner provide published information concerning these assertions. This is because the § 103 rejections are apparently being based on assertions that draw on facts within the personal knowledge of the Examiner, since no support was provided for these otherwise conclusory and unsupported assertions. (See also MPEP § 2144.03).*

It is therefore respectfully submitted that claims 1 to 17 are allowable.

## CONCLUSION

It is therefore respectfully submitted that all of the presently pending claims are allowable. It is therefore respectfully requested that the rejections (and any objections) be withdrawn, since all issues raised have been addressed and obviated. An early and favorable action on the merits is respectfully requested.

Respectfully submitted,

Dated: 4/8/2009

By: _____

Gerard A. Messina
Reg. No. 35,952
KENYON & KENYON LLP
One Broadway
New York, New York 10004
(212) 425-7200

**CUSTOMER NO. 26646**

1690819